1. (6 pts) Fill in the blanks in the following piece of event handling code that handles clicks on a JButton.

```java
JButton btn = new JButton ("Click here");

btn.addActionListener (

    _new_ _ActionListener________()

    {
        public void _actionPerformed (_ActionEvent_ e)
        {
            System.out.println ("Button pressed.");
        }
    });

public interface ActionListener
{
    public void actionPerformed (ActionEvent e);
}

public interface MouseListener
{
    public void mouseClicked(MouseEvent e);
    public void mouseEntered(MouseEvent e);
    public void mouseExited(MouseEvent e);
    public void mousePressed(MouseEvent e);
    public void mouseReleased(MouseEvent e);
}

public class MouseAdapter
{
    public void mouseClicked(MouseEvent e);
    public void mouseEntered(MouseEvent e);
    public void mouseExited(MouseEvent e);
    public void mousePressed(MouseEvent e);
    public void mouseReleased(MouseEvent e);
}
```
2. (8 pts) For the following interface, indicate what components would be needed to construct it. Include "invisible" components such as panels if they are needed for layout.

(Note: Other designs are possible.)

3. (6 pts) Why are GUI programmers in Java instructed to avoid putting code in event handlers that will take a long time to run?

- only handle one event at a time
- no other interface actions will be handled during EH execution

Extra credit: (2 pts) What layout manager(s) would be needed to organize the components in Question #2?

In the layout above, all of the containers would have a Box layout.

(Note: other designs would require different layout choices.)